AMENDMENTS TO THE CLAIMS

1.   (Currently Amended) A database system comprising:

processors;

a storage subsystem having plural storage modules to store respective portions of a view containing results of a cube-based operation on at least one base table, the view containing a first result set for a group-by on a first grouping set, and a second result set for a group-by on a second grouping set, wherein the second grouping set has less attributes than the first grouping set; and

database software executable on the processors and having plural access modules that provide parallel access of data in the plural storage modules, wherein the database software executable on the processors is configured to:

distribute the portions of the view across the plural storage modules based on hash values computed by hashing plural attributes specified by a function defining the view;

in response to a change to the at least one base table:

update the first result set by computing a change to the first result set based on the change in the at least one base table;

distribute rows of the change to the first result set across the plural storage modules based on hash values computed by hashing the attributes of the second grouping set and at ~~lest~~ least one other attribute that is part of the first grouping set but not the second grouping set, wherein the at least one other attribute is assigned a predefined value <u>used by said database software executable</u> to perform the hashing of the attributes of the second grouping set and the at least one other attribute; and

update the second result set by computing a change to the second result set based on the distributed rows of the change to the first result set.

2.   (Original) The database system of claim 1, wherein the first grouping set has a first number of grouping attributes, and the second grouping set has a second number of grouping attributes, the first number being greater than the second number.

3.   (Cancelled)

4.   (Original) The database system of claim 1, wherein the view contains results of a group-by cube operation.

5.   (Original) The database system of claim 1, wherein the view contains results of a group-by partial cube operation.

6.-7.   (Cancelled)

8.   (Previously Presented) The database system of claim 1, wherein the function is a cube function, and the view contains results of a cube operation specified by the cube function on the plural attributes, and wherein the at least another attribute is one of the plural attributes of the cube function that is not in the second grouping set.

9.   (Currently Amended) The database system of claim 1, wherein the rows of the change to the first result set are distributed across the access modules according to the hashing of the attributes of the second grouping set and the at least one other attribute to enable enabling:

        each access module to locally perform a merge and aggregate operation on the rows of change to the first result set to update the second result set; and

each access module to locally merge the rows of the change to the second result set into a respective portion of the view without having to first redistribute the rows of the change to the second result set.

10.   (Currently Amended) The database system of claim 1, wherein the database software is configured to further:

receive a query specifying a group-by operation; and

determine whether an answer for the query specifying the group-by operation ~~can be~~ is satisfied from the view.

11.   (Previously Presented) The database system of claim 10, wherein the query specifies a group-by operation on grouping sets S, and the view contains result sets for grouping sets C,

the database software being configured to determine whether S is a subset of C to determine whether the answer for the query can be satisfied from the view.

12.   (Previously Presented) The database system of claim 11, wherein the database software is configured to modify a WHERE clause of the query in response to determining that S is a subset of C.

13.   (Currently Amended) A method for use in a database system, comprising:

storing a view containing results of a cube-based operation on at least one base table, the view containing result sets for group-bys on respective grouping sets, wherein portions of the view are distributed across plural storage modules of a storage subsystem, wherein the result sets include a first result set and a second result set, wherein the first result set is for a group-by on a first of the grouping

sets, and the second result set is for a group-by on a second of the grouping sets, and wherein the second grouping set has less attributes than the first grouping set;

accessing data in the plural storage modules in parallel by corresponding plural access modules in database software executing on processors;

distributing, by the processors, the portions of the view across the plural storage modules based on hash values computed by hashing plural attributes specified by a function defining the view:

updating, by the processors, the first result set by computing a change to the first result set based on a change in the at least one base table;

distributing, by the processors, rows of the change to the first result set across the plural storage modules based on hash values computed by hashing the attributes of the second grouping set and at least one other attribute that is part of the first grouping set but not the second grouping set, wherein the at least one other attribute is assigned a predefined value <u>used by said processors</u> to perform the hashing of the attributes of the second grouping set and the at least one other attribute; and

updating, by the processors, the second result set by computing a change to the second result set based on the distributed rows of the change to the first result set.

14.   (Cancelled)

15.   (Original) The method of claim 13, further comprising updating a third result set by computing a change to the third result set based on the change to the second result set.

16.   (Original) The method of claim 15, further comprising updating a fourth result set by computing a change to the fourth result set based on the change to the third result set.

17.-18. (Cancelled)

19.   (Previously Presented) The method of claim 13, wherein the first result set corresponds to the group-by on the first grouping set having N attributes, and the second result set corresponds to the group-by on the second grouping set having N-1 attributes, and wherein the N-1 attributes in the second group set are all part of the first grouping set,

20.   (Previously Presented) The method of claim 19, wherein the result sets of the view further comprise a third result set for a group-by on a third of the grouping sets, wherein the third grouping set has less attributes than the second grouping set, the method further comprising:

distributing rows of the change to the second result set across the plural storage modules based on hash values computed by hashing the attributes of the third grouping set and at least two other attributes that are part of the first grouping set but not the third grouping set, wherein the at least two other attributes are assigned the predefined value to perform the hashing of the attributes of the third grouping set and the at least two other attributes;

updating the third result set by computing a change to the third result set based on the distributed rows of the change to the second result set.

21.   (Previously Presented) The method of claim 20, wherein the function is a cube function, and wherein storing the view comprises storing the view for a cube operation based on the cube function of the N attributes.

22.   (Currently Amended) An article comprising at least one storage medium containing instructions that when executed cause a database system to:

store a view containing results of a cube-based operation on at least one base table, the view containing result sets for group-bys on respective grouping sets, wherein portions of the view are distributed across plural storage modules of a storage subsystem, wherein the result sets include a first result set and a second result set, wherein the first result set is for a group-by on a first of the grouping sets, and the second result set is for a group-by on a second of the grouping sets, and wherein the second grouping set has less attributes than the first grouping set;

access data in the plural storage modules in parallel by corresponding plural access modules in database software in the database system;

distribute the portions of the view across the plural storage modules based on hash values computed by hashing plural attributes specified by a function defining the view;

update the first result set by computing a change to the first result set based on a change in the at least one base table; and

distribute rows of the change to the first result set across the plural storage modules based on hash values computed by hashing the attributes of the second grouping set and at least one other attribute that is part of the first grouping set but not the second grouping set, wherein the at least one other attribute is assigned a predefined value used by said database system in the execution of the instructions to perform the hashing of the attributes of the second grouping set and the at least one other attribute; and

update the second result set by computing a change to the second result set based on the distributed rows of the change to the first result set.

23.   (Cancelled)

24.   (Original) The article of claim 22, wherein the instructions when executed cause the database system to further update a third result set by computing a change to the third result set based on the change to the second result set.

25.-26. (Cancelled)

27.   (Previously Presented) The article of claim 13, wherein the first result set corresponds to the group-by on the first grouping set having N attributes, and the second result set corresponds to the group-by on the second grouping set having N-1 attributes, and wherein the N-1 attributes in the second group set are all part of the first grouping set.

28.   (Previously Presented) The article of claim 27, wherein the function is a cube function, and wherein storing the view comprises storing the view for a cube operation based on the cube function of the N attributes.

29.   (Previously Presented) The database system of claim 1, wherein the view further contains a third result set for a group-by on a third grouping set having less attributes than the second grouping set, and wherein the database software is configured to further:

distribute rows of the change to the second result set across the plural storage modules based on hash values computed by hashing the attributes of the

third grouping set and at least two other attributes that are part of the first grouping set but not the third grouping set, wherein the at least two other attributes are assigned the predefined value to perform the hashing of the attributes of the third grouping set and the at least two other attributes; and

update the third result set by computing a change to the third result set based on the distributed rows of the change to the second result set.

30.   (Previously Presented) The database system of claim 29, wherein all of the attributes of the second grouping set are part of the first grouping set, and all of the attributes of the third grouping set are part of the second grouping set.

31.   (Previously Presented) The article of claim 22, wherein the result sets of the view further include a third

result set for a group-by on a third of the grouping sets having less attributes than the second grouping set, and wherein the instructions when executed cause the database system to further:

distribute rows of the change to the second result set across the plural storage modules based on hash values computed by hashing the attributes of the third grouping set and at least two other attributes that are part of the first grouping set but not the third grouping set, wherein the at least two other attributes are assigned the predefined value to perform the hashing of the attributes of the third grouping set and the at least two other attributes; and

update the third result set by computing a change to the third result set based on the distributed rows of the change to the second result set.